MIKE'S NOTEBOOK

By: Michael J. Di Rienzo

(NOTE: REPRINTING OR REPRODUCING THIS COLUMN WITHOUT THE EXPRESSED WRITTEN PERMISSION OF THE AUTHOR IS HEREBY PROHIBITED. FOR PERMISSION, WRITE THE AUTHOR IN CARE OF THIS PUBLICATION.)

Here's another easy-to-use command utility which quickly PRINTs a string of text at any size and at any pixel coordinate location. You will be able to use this utility in your own programs. The syntax follows: PRINT USR large;AT X,Y;W,H;"txt" The text at the end of the line can either by within quotes or use a string variable (A$...Z$). The variables "X" & "Y" are the column (0-255) and line (0-191) pixel location where you want the text to commence PRINTing. The coordinate 0,0 is at the upper left corner of the screen. The "W" variable is the letter width in terms of characters (1-32), and "H" is the height of the letter (1-24). Be sure to note the commas and semicolons in the command syntax. If you make an error, you will get a "Nonsense in Basic" error. Note that the PAPER and INK command can be used within the PRINT statement in their usual way. FLASH also works. If your text runs off the screen, it will wrap around to the opposite side, so you'll have to plan for text length and size. Be sure to RUN line 100 to initialize the CODE before RUNning for the first time, then use RUN thereafter. The CODE is not relocatable and is 350 bytes long. Have fun!

Happy TIMEXing...

"LARGE"

By Michael J. Di Rienzo

```
  10 BORDER 0: PAPER 0: INK 7: C
LS : OUT 254,1: LET large=60400
  20 LET d$="2068 LARGE print"
  30 RANDOMIZE
  40 LET w=INT (1+RND*2)
  50 LET h=INT (1+RND*12)
  60 LET x=INT (RND*(256-(w*8*16
)))
  70 LET y=INT (RND*(192-(h*8)))
  80 LET i=INT (1+RND*6)
  90 PRINT INK i;USR large;AT x,
y;w,h;d$: PAUSE 100: CLS : GO TO
 40
```

```
 100 CLEAR 59399: LET t=0
 110 FOR n=60400 TO 60749
 120 READ a: POKE n,a: LET t=t+a
 130 NEXT n: IF t<>31728 THEN PR
INT FLASH 1;"Data Error!": PRINT
 "Recheck DATA lines": STOP
 140 RUN
 150 DATA 17,117,33,225,167,237,
82,32,250,213,223,254,59,32,40,2
31,254,172,32,35,205,220,27,205,
96,38,237,67,10,91,223,254,59,32
,20,205,220,27,205,96,38,237,67,
12,91,62,8,50,14,91,223,254,59,4
0,2,207,11,231,205,239,27,205,17
5,47,33,15,91,235,237,176,62,255
,18
 160 DATA 33,15,91,126,35,34,0,9
1,111,60,200,38,0,41,41,41,237,7
5,54,92,9,62,8,50,4,91,58,11,91,
50,9,91,58,10,91,50,8,91,62,9,50
,5,91,126,35,34,2,91,7,50,6,91,5
8,5,91,61,32,50,58,4,91,61,32,24
,58,14
 170 DATA 91,71,58,12,91,79,58,1
0,91,129,5,32,252,50,10,91,42,0,
91,195,60,236,50,4,91,58,13,91,7
1,58,9,91,128,50,9,91,42,2,91,19
5,89,236,50,5,91,58,12,91,71,58,
9,91,50,7,91,58,13,91,79,197,205
,221,236,193,58,7
 180 DATA 91,60,50,7,91,13,32,24
1,58,8,91,60,50,8,91,5,32,221,58
,6,91,195,105,236,128,64,32,16,8
,4,2,1,58,142,92,238,255,71,58,1
43,92,160,71,58,8,91,230,248,111
,58,7,91,254,192,208,31,31,31,23
0,31,103,203,28,203,29,203,28,20
3,29,203,28,203,29,62,88,180
 190 DATA 103,58,142,92,166,176,
119,58,7,91,71,230,7,246,64,103,
120,31,31,31,230,24,180,103,120,
23,23,230,224,111,58,8,91,71,31,
31,31,230,31,181,111,235,33,213,
236,120,230,7,79,6,0,9,70,26,33,
6,91,203,70,40,3,176,18,201,47,1
76,47,18,201
```

MIKE'S NOTEBOOK

By: Michael J. Di Rienzo

(NOTE: THIS COLUMN MAY NOT BE REPRINTED WITHOUT THE EXPRESS WRITTEN PERMISSION OF THE AUTHOR. TO OBTAIN PERMISSION, WRITE TO THE AUTHOR IN CARE OF RMG ENTERPRISES.)

I like to work with computer graphics on the TS2068 and I frequently need to look up bit patterns for User Defined Graphics characters. Knowing the bit patterns of a specific byte can also help to better understand machine code among the many other uses. The following BASIC program will print out all 256 bit patterns, their hexadecimal and decimal equivalent. A zero is represented as a blank box and a one is a filled box. In fact, the UDG "A" and "B" are poked with their pre-planned bit patterns in lines 300 to 340. To see what I mean, just type in the following program which will fill up the page then prompt you to Quit, COPY to 2040 printer or CONTINUE. If you want to copy to full sized printers then change line 240 and add your own printer driver to COPY the screen. I always have a copy of all the bit patterns handy while programming. I added line 115 and changed line 120 to increase the pattern print speed by 65%. Make sure that you use UDG graphics "A" in line 140 and UDG "B" in 150. Lines 300 to 340 POKE the UDG's with the proper bytes.

Happy TIMEXing...

"BITS" By Michael Di Rienzo

```
 10 GO SUB 300
 20 POKE 23658,8: LET n=0
 30 LET h$="0123456789ABCDEF"
 40 FOR c=1 TO 2
 50 IF c=1 THEN LET c1=0: LET c
2=4: LET c3=7
 60 IF c=2 THEN LET c1=17: LET
c2=21: LET c3=24
 70 PRINT AT 0,c1;"Dec";TAB c2;
"Hex";TAB c3+1;"Binary"
 80 FOR z=1 TO 21
 90 LET s=INT (n/16)
100 PRINT AT z,c1;n;TAB c2;h$(1
+s);h$(1+n-16*s);TAB c3;
110 FOR b=7 TO 0 STEP -1
115 LET i=1: FOR w=1 TO b: LET
1=1*2: NEXT w
120 LET x=INT (n/1)
130 LET t=x-INT (x/2)*2
140 IF t=1 THEN PRINT "A";
150 IF t=0 THEN PRINT "B";
160 NEXT b
170 PRINT
180 LET n=n+1
190 IF n=256 THEN GO TO 220
200 NEXT z
210 NEXT c
```

```
 220 PRINT #1;AT 1,0;"(C)opy,(Q)
uit,ENTER to Continue"
 230 LET q=CODE INKEY$
 240 IF q=67 THEN COPY : LPRINT
: IF n<256 THEN CLS : GO TO 40
 250 IF q=81 THEN STOP
 260 IF q=13 AND n=256 THEN STOP
 270 IF q=13 THEN CLS : GO TO 40
 280 GO TO 230
 300 FOR n=USR "A" TO USR "A"+15
 310 READ a: POKE n,a
 320 NEXT n
 330 RETURN
 340 DATA 0,0,126,126,126,126,12
6,0,0,0,126,66,66,66,126,0
```

MIKE'S NOTEBOOK

By: Michael J. Di Rienzo

(NOTE: REPRINTING OR REPRODUCING THIS COLUMN WITHOUT THE EXPRESSED WRITTEN PERMISSION OF THE AUTHOR IS HEREBY PROHIBITED. FOR PERMISSION, WRITE THE AUTHOR IN CARE OF THIS PUBLICATION.)

This months utility gives you a new WINDOW command that will place a user defined 3-D window at any location and any size on the screen using any of the eight available colors. The user canl define the line, column, width, height, and color of each window. When using this utility in your own home-grown programs, WINDOW will provide an appropriate background to print text, prompts, or instructions. If used together with my GETPUT utility published previously, you can SAVE, then replace the area of the screen covered-up by the window. This program provides a graphic background only and doesn't open a new channel, so PRINT will work in its usual way. Error trapping is minimal for incorrectly sized windows, but is provided for the BASIC syntax. The syntax of this new command must be followed correctly or you will get a "Nonsense in Basic" error message. Windows that go off the screen will generate an "Integer out of range" error message. The syntax is as follows: PRINT USR window;AT L,C;W,H;A Note the commas and semi-colons. Variable "L" is the line (1-20) and "C" is the column (1-30) where you want the upper left corner of the window to appear. Variable "W" is the width (1-30) and "H" is the height (1-20) (in characters) of the window that you want to define. "A" is the attribute color (0-7) that you want to paint the window. The window shadow is always black. The CODE must be initialized by RUNning line 200 first. The CODE is not relocatable and is 323 bytes long. Study the following demo to understand your new utility command. Have fun!

Happy TIMEXing...

"WINDOW" By Michael J. Di Rienzo

```
  10 LET window=49856
  20 PRINT USR window;AT 5,4;25,
4;5
  30 PRINT PAPER 5; INK 2;AT 6,5
;"This is the window DEMO!"; FLA
SH 1;AT 7,5;"Press a key to cont
inue."
  40 PAUSE 0
  50 LET line=INT (1+RND*19)
  60 LET column=INT (1+RND*29)
  70 LET width=INT (1+RND*(31-co
lumn))
  80 LET height=INT (1+RND*(21-1
ine))
  90 LET color=INT (1+RND*7)
 100 PRINT USR window;AT line,co
lumn;width,height;color
 110 GO TO 50
 200 CLEAR 49855: LET t=0
 210 FOR n=49856 TO 50178
 220 READ a: POKE n,a: LET t=t+a
```

```
 230 NEXT n: IF t<>32454 THEN PR
INT FLASH 1;"Data Error! Recheck
 DATA lines": STOP
 240 RUN
 250 DATA 17,117,33,225,167,237,
82,32,250,213,223,254,59,32,35,2
31,254,172,32,30,205,220,27,205,
96,38,237,67,0,195,223,254,59,32
,15,205,220,27,205,96,38,237,67,
2,195,223,254,59,40,2,207,11,205
,228,27,205,96,49,121,50,13,195
 260 DATA 24,4,0,0,0,0,62,2,205,
48,18,62,17,215,62,3,215,237,91,
0,195,58,3,195,71,58,2,195,79,62
,22,215,123,215,122,215,62,32,21
5,13,32,250,28,16,236
 270 DATA 253,54,86,56,33,22,0,2
37,91,0,195,205,69,23,121,50,0,1
95,58,0,195,135,135,135,50,0,195
,58,1,195,135,135,135,61,50,1,19
5,58,2,195,135,135,135,60,50,2,1
95,58,3,195,135,135,135,60,50,3,
195
 280 DATA 58,1,195,79,58,0,195,7
1,205,62,38,58,2,195,79,6,0,17,1
,1,205,163,195,14,0,58,3,195,71,
30,1,22,255,205,163,195,58,2,195
,79,6,0,30,255,22,1,205,163,195,
14,0,58,3,195,71,17,1,1,205,163,
195,24,10,217,229,217,205,19,40,
217,225,217,201
 290 DATA 58,3,195,71,58,0,195,1
52,61,71,46,3,229,58,1,195,60,60
,60,79,197,205,62,38,58,2,195,79
,6,0,17,1,1,205,163,195,193,5,22
5,45,32,226,58,1,195,71,58,2,195
,128,60,79,46,3,229,58,0,195,61,
61,61,71,197,205,62,38,14,0,58,3
,195,71,22,255,30,1,205,163,195,
193,12,225,45,32,225,201
```

(EDITOR'S NOTE: If you enjoy this column and would like to see it continue, PLEASE let the auther know! If we do not know that you, the readers, like or can use his work,why should he continue? LET'S HEAR FROM YOU!)

MIKE'S NOTEBOOK

By: Michael J. Di Rienzo

(NOTE: REPRINTING OR REPRODUCING THIS COLUMN WITHOUT THE EXPRESS WRITTEN PERMISSION OF THE AUTHOR IS HEREBY PROHIBITED. FOR PERMISSION, WRITE THE AUTHOR IN CARE OF RMG ENTERPRISES.)

My friends often encourage me to put my powerful TS2068 ROM to work for a good cause. What better way than to win the state lottery. Now, I know that lotto number generators are ubiquitous but this one really works (fast). It can be modified for your geographical requirements. All you need to know is the range of numbers possible and the maximum number of picks per game. In my state, the player specifies six (6) numbers ranging from 1 to 44. To change game parameters, type in the following program and make the following changes: The range of numbers (1 to 44) are specified in lines 40,60, and 120: The number of picks (6) are found in lines 50 and 130. Line 20 asks how many lines you would like to choose. Just follow the prompt and you will be presented with one to ten extremely lucky numbers. GOOD LUCK!!! You may forward me 10% of all your earnings and be assured that you are contributing to a good cause.

Happy TIMEXing...

```
  "LOTTO" By Michael Di Rienzo

   1 BORDER 4: PAPER 0: CLS : PA
PER 1: INK 6
   2 FOR x=0 TO 31 STEP 2
   3 PRINT FLASH 1;AT 0,x;CHR$ 1
43;CHR$ 128;AT 21,30-x;CHR$ 128;
CHR$ 143
   4 IF x<22 THEN PRINT FLASH 1;
AT x,0;CHR$ 143;AT x+1,0;CHR$ 12
8;AT 20-x,31;CHR$ 128;AT 20-x+1,
31;CHR$ 143
   5 NEXT x
  10 RANDOMIZE : PAPER 0: INK 7
  20 INPUT "How Many Lines (1-10
)? ";q: IF q>10 THEN GO TO 20
  30 FOR g=1 TO q
  40 DIM s(44)
  50 FOR n=1 TO 6
  60 LET a=1+INT (RND*44)
  70 IF NOT s(a) THEN LET s(a)=a
: GO TO 90
  80 GO TO 60
  90 NEXT n
 100 PRINT AT g*2,5;"(";g;") ";
 110 LET c=0
 120 FOR n=1 TO 44
 130 IF s(n) THEN LET c=c+1: PRI
NT s(n);"," AND c<6;
 140 NEXT n
 150 BEEP .01,20: PRINT ''
 160 NEXT g: PRINT #1;AT 1,10;"G
ood Luck!!"
 170 PAUSE 0: STOP
```

MIKE'S NOTEBOOK

By: Michael J. Di Rienzo

(NOTE: REPRINTING OR REPRODUCING THIS COLUMN WITHOUT THE EXPRESSED WRITTEN PERMISSION OF THE AUTHOR IS HEREBY PROHIBITED. FOR PERMISSION, WRITE THE AUTHOR IN CARE OF THIS PUBLICATION.)

Sinclair BASIC graphics commands PLOT and DRAW have several disadvantages and shortcomings. One of them is that they only address a 256 X 176 pixel screen and cannot draw to the bottom two character lines. Another is that the PLOT command requires absolute coordinate pairs but the DRAW command needs relative offset pairs. This requires a conversion somewhere in the program. This months utility will provide three graphics commands (PLOT, DRAW, and LINE) that will draw to the entire 256 by 192 pixel screen, and all x,y coordinate pairs required are the absolute type. The other change is that the coordinate Ø, Ø is at the upper-left corner of the screen. The command syntax is as follows: PRINT USR plot;AT xs,ys PRINT USR draw;AT xe,ye PRINT USR line;AT xs,ys;xe,ye All "X" values are Ø to 255, and all "Y" values are Ø to 191. The variables xs=x start, ys=y start, xe=x end, and ye=y end. As usual, DRAW requires that it be preceeded by PLOT which establishes the starting x,y coordinate. Next, Draw sets the end coordinate pair and connects them with a line. The new command, LINE, requires two coordinate pairs. The first pair sets the x,y start of the line and the second pair sets the end x,y pair and connects them with a line. Study the included example program to see how to use them. Note that the syntax of the new commands is important in terms of commas and semicolons. If you make a syntax error with the new commands, you will get a "Nonsense in Basic" error message. Note that you may use attribute commands in the usual way because each command is called using, "PRINT USR...". After the design is drawn and the border is plotted, press any key to draw another demo. Be sure to RUN the program from line 3ØØ to POKE in the CODE. You can use RUN after that. The CODE is not relocatable and is 249 bytes long if you want to SAVE it as a separate CODE file. A portion of the plot and draw routines were obtained from the "Advanced 2Ø68 Machine Code", Volume 1, by Dr. Lloyd Dreger, and were enhanced a bit. I obtained this resource and others by Dr. Dreger from RMG ENTERPRISES. Have fun!

Happy TIMEXing...

"LINES"

By Michael J. Di Rienzo

```
  1Ø RANDOMIZE : BORDER Ø: PAPER
 Ø: INK 1+RND*6: CLS : OUT 254,1
: LET plot=6Ø987: LET line=6Ø99Ø
: LET draw=6Ø994
  2Ø LET m=(RND*2Ø)+3Ø
  3Ø FOR e=1Ø TO 6Ø STEP .5
  4Ø LET r=e*88/m
  5Ø LET t=PI*1ØØ*e/m
  6Ø LET xe=128+1.2*r*COS t
  7Ø LET ye=96+r*SIN t
  8Ø IF ye<Ø OR ye>191 OR xe<Ø O
R xe>255 THEN GO TO 12Ø
```

```
 90 IF e=10 THEN GO SUB 250
 100 GO SUB 240
 110 NEXT e
 120 FOR n=2 TO 253 STEP 2: PRIN
T USR plot;n,2: PRINT USR plot;n
,189: PRINT USR plot;2,n AND n<=
189: PRINT USR plot;253,n AND n<
=189: NEXT n
 130 PRINT USR line;0,0;255,0: P
RINT USR line;255,0;255,191: PRI
NT USR line;255,191;0,191: PRINT
 USR line;0,191;0,0
 140 PAUSE 0: RUN
 240 PRINT USR PLOT;xs,ys: PRINT
 USR DRAW;xe,ye
 250 LET xs=xe: LET ys=ye
 260 RETURN
 300 CLEAR 60820: LET t=0
 310 FOR n=60821 TO 61069
 320 READ a: POKE n,a: LET t=t+a
 330 NEXT n: IF t<>29525 THEN PR
INT FLASH 1;"Data Error! Recheck
 DATA lines": STOP
 340 RUN
 350 DATA 125,230,192,31,55,31,1
5,173,230,248,173,87,124,7,7,7,1
73,230,199,173,7,7,95,213,122,15
,15,15,230,3,246,88,87,237,75,14
3,92,26,169,160,169,18,209,124,2
30,7,71,4,62,254,15,16,253,71,58
,145,92,79,26,203,65,32,1,160,20
3,81,32,2,168,47,18,201
 360 DATA 1,0,1,0,197,205,149,23
7,193,17,1,1,120,148,48,4,21,21,
237,68,71,121,149,48,4,29,29,237
,68,79,176,200,121,184,229,98,10
7,34,221,237,46,0,56,4,101,107,7
2,71,34,223,237,96,120,203,63,11
1,125,129,56,3,184,56,8,144,111,
237,91,221,237,24,5,111,237,91,2
23,237,227,124,130,103,125,131,1
11,197,205,149,237,193,227,37,32
,220,225,201
 370 DATA 175,24,6,62,1,24,2,62,
2,50,176,92,17,117,33,225,167,23
7,82,32,250,213,58,176,92,254,2,
32,9,237,75,125,92,97,104,195,11
3,238,205,125,238,97,104,58,176,
92,254,1,40,4,205,149,237,201,22
9,205,125,238,120,65,79,225,205,
225,237,201,223,254,59,194,237,2
7,205,220,27,205,96,38,237,67,12
5,92,201
```

MIKE'S NOTEBOOK

By: Michael J. Di Rienzo

(NOTE: REPRINTING OR REPRODUCING THIS COLUMN WITHOUT THE EXPRESSED WRITTEN PERMISSION OF THE AUTHOR IS HEREBY PROHIBITED. FOR PERMISSION, WRITE THE AUTHOR IN CARE OF THIS PUBLICATION.)

If you are used to SAVEing SCREEN$ files, you may have noticed that the file size is rather large especially if you LOAD or SAVE using cassette storage medium. The file is 6912 bytes long. Another problem to overcome is that if you want to store more than four or five SCREEN$'s in RAM memory, there is hardly any memory left for your programs. This months utility will COMPRESS a typical SCREEN$ file up to as much as 1/2 its usual size. In addition, a header is automatically included in the SAVED file which will resolve the compressed SCREEN$ back to its full size onto the screen by simply using RANDOMIZE USR (address). The following program will SAVE the compressed SCREEN$ CODE at address 40246 and you will be shown the file length if you wish to write it down. When you are ready to LOAD the compressed file into your own program, you can LOAD it to any legal address then "call" that address to get your expanded SCREEN$. The following program will POKE in the CODE then prompt you to INPUT the file name of the SCREEN$ that you want to compress. It will then LOAD it and temporarily store it. Next you'll be asked what name you want to give the compressed SCREEN$. The SCREEN$ will be instantly compressed and you will be notified of the start address and byte length of the file. Now, a prompt to press any key when you are ready to SAVE your file. Don't worry, your compressed file is already stored at address 40246. This utility can be easily modified by you to work with any disk system.

I use this utility alot to animate as many as 15 compressed SCREEN$ by LOADing them consecutively in memory then calling them in sequence to simulate animation. The CODE is not relocatable and is 132 bytes long. Have fun!

Happy TIMEXing...

"CMPRES"

By Michael J. Di Rienzo

```
  10 CLEAR 36611: LET store=
36720: LET recall=36732: LET
compress=36612: GO SUB 100
  20 INPUT "What is the file
name to LOAD? ";n$
  30 LOAD n$SCREEN$ : RAND
OMIZE USR store
  40 INPUT "What is the file
name to SAVE? ";n$: RANDOMIZE
USR recall
  50 RANDOMIZE USR compress:
LET L=USR 40192: LET L=L-40246
  60 PRINT INK 2; PAPER 5; AT
10, 0;"File:""";n$;"""CODE
40246, ";L ; FLASH 1;AT 11,5;
"Press any key to SAVE": PAUSE
0
  70 SAVE n$CODE 40246,L
  80 CLS : RANDOMIZE USR 40246
: PRINT INK 1; FLASH 1; AT 10,
0;"To resolve the compressed f
ile back to the screen in the
future RANDOMIZE USR (Start ad
dress where CODE is LOADed!") :
  90 STOP
 100 LET t=0
 110 FOR N=36612 TO 36743
 120 READ a: POKE n,a: LET t=t+a
 130 NEXT n: IF t<>12070 THEN PRI
NT FLASH 1;"Data Error! Recheck D
ATA line.": STOP
 140 RETURN
 150 DATA 17,0,157,33,16,143,1,98
,0,237,176,201,197,209,33,100,0,2
5,235,33,0,64,14,1,126,167,40,9,2
54,255,40,5,237,160,3,24,21,71,12
1,60,35,40,7,126,184,32,3,12,24,2
43,120,18,19,121,18,19,14,1,124,2
54,91,32,217,213,193,201,17,0,64,
221,33,46,0,221,9,122,254,91,200,
221,126,0,167,40,10,254,255,40,6,
18,19,221,35,24,236,221,70,1,18,1
9,16,252,221,35,221,35,24,223
 160 DATA 17,0,157,33,0,64,1,0,24
0,237,176,201,17,0,64,33,0,157,1,0,
24,237,176,201
```